

(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開平6-183062

(43)公開日 平成 6年(1994) 7月 5日

(51)Int.Cl. ⁵	識別記号	庁内整理番号	F I	技術表示箇所
B 4 1 J	2/485			
	5/44	8703-2C		
G 0 9 G	5/22	8121-5G		
		8703-2C	B 4 1 J 3/ 12	C

審査請求 未請求 請求項の数 1 (全 13 頁)

(21)出願番号 特願平4-353891

(22)出願日 平成 4年(1992)12月16日

(71)出願人 000005496

富士ゼロックス株式会社

東京都港区赤坂三丁目 3 番 5 号

(72)発明者 鹿島 洋三

埼玉県岩槻市府内 3 丁目 7 番 1 号 富士ゼ

ロックス株式会社内

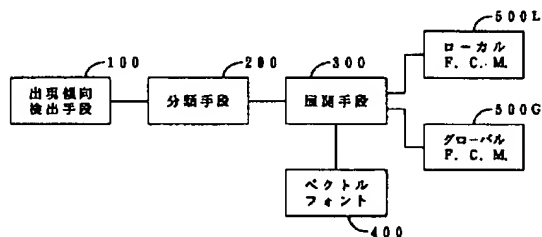
(74)代理人 弁理士 平木 道人 (外 1 名)

(54)【発明の名称】 文字処理装置

(57)【要約】

【目的】 フォントキャッシュメモリの使用効率を大幅に向上することができる文字処理装置を提供すること。

【構成】 出現傾向検出手段 1 0 0 は、文書の予定の範囲を対象に、各文字の出現傾向を予め検出する。分類手段 2 0 0 は、前記出現傾向に基づいて、各文字を、文書中の狭い範囲で出現するローカル文字および広範囲にわたって出現するグローバル文字に分類する。展開手段 3 0 0 は、印字要求のあった文字をベクトルフォント 4 0 0 から読み出して展開し、ローカル文字はローカルフォントキャッシュメモリ 5 0 0 L へ登録し、グローバル文字はグローバルフォントキャッシュメモリ 5 0 0 G へ登録する。



【特許請求の範囲】

【請求項1】ベクトルフォントをドットパターンに展開して登録するフォントキャッシュメモリを備え、未登録データはベクトルフォントから展開し、既登録データはフォントキャッシュメモリから読み出す文字処理装置において、

文書の予定の範囲を対象に、各文字の出現傾向を予め検出する手段と、

前記出現傾向に基づいて、各文字を局部的出現文字および全体的出現文字に分類する手段と、

局部的出現文字のドットパターンを記憶する第1のフォントキャッシュメモリと、

全体的出現文字のドットパターンを記憶する第2のフォントキャッシュメモリとを具備したことを特徴とする文字処理装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は文字処理装置に係り、特に、ベクトル形式で表された文字等の図形データ（ベクトルフォント）をドット形式のデータに展開して登録するフォントキャッシュメモリを備えた文字処理装置に関する。

【0002】

【従来の技術】従来、日本語ワードプロセッサ等におけるプリント装置においては、 $24 \times 24 \sim 56 \times 56$ ドット程度の点の集まりで一つの文字パターンを表すドットフォントか、ベクトルデータを組合わせることにより一つの文字を表すベクトルフォントのどちらかを採用している。

【0003】そのうち、ベクトルフォントは、ドットフォントに比較して、文字を大きく拡大しても輪郭がなめらかで綺麗に印字できるという利点があるため、多くのプリント装置で採用されている。

【0004】ベクトルフォントにおいては、ある印字要求があった時、その文字コードに対応するベクトルデータをベクトルデータ記憶部から読み出した後、ベクトル演算を行ってフォントの輪郭を描き、輪郭内部を塗り潰すことにより、ドットパターンを生成することができる。

【0005】このようにして生成されたベクトルフォントのドットパターンは、フォントキャッシュに記憶される。つまり、フォントキャッシュメモリは、ベクトル演算、輪郭作成、塗り潰しによって生成されたベクトルフォントのドットパターンを記憶するメモリである。

【0006】生成されたドットパターンは、次々に、フォントキャッシュメモリに格納される。もし、新たに印字したい文字がフォントキャッシュメモリ内に既に存在すれば、新たにベクトル演算、輪郭作成、塗り潰し作業を行う必要がなく、フォントキャッシュメモリ内の文字のドットパターンを使って印字することができるため、

印字速度が向上する。

【0007】理想的には、あらゆるフォントを次々にフォントキャッシュメモリに格納すれば、最終的には、ドットパターンを生成する必要がなくなり、全てのフォントをフォントキャッシュメモリ内の文字のドットパターンを使って印字することができるため、印字速度はドットフォントを使った場合と同等になる。

【0008】しかし、フォントキャッシュメモリの記憶領域は有限であるため、使われなくなったドットパターンは追出して、新しいドットパターンを格納するという、ドットパターンの入替えを行わなければならない。

【0009】

【発明が解決しようとする課題】従来は、どのドットパターンが必要でないかを判定するために、FIFO方式、LRU方式などの方式が一般的に使用されている。

【0010】FIFO (First in First out) 方式フォントキャッシュは、フォントキャッシュメモリに格納された時刻の古いものから先に追出していく方式であり、簡単に実現できるというメリットがある。この方式を開示している公報として、例えば特開平3-48286号公報「文字処理系におけるフォントキャッシュ制御方式」がある。

【0011】しかしながら、使用頻度が高い文字であっても、フォントキャッシュメモリに格納された時刻が古くなれば、フォントキャッシュメモリから追出されてしまうので、フォントキャッシュメモリ内のドットパターンの再利用率（すなわち、フォントキャッシュメモリ内に格納されているデータが再び利用される割合）が低いという問題がある。

【0012】LRU (Least Recently Used) 方式フォントキャッシュは、フォントキャッシュメモリに格納されているもののうち、使用された時刻の古いものから先に追出されていく方式であるので、必然的に使用頻度の低いものから順に追出されていくことになる。よって、フォントキャッシュメモリ内には、使用頻度の高いものが残されることになり、ドットパターンの再利用率は向上するというメリットがある。

【0013】この方式を開示している公報として、例えば特開昭61-190382号公報「文字フォント管理方式」、特開昭61-230189号公報「文字パターンローディング方式」、および特開平3-200196号公報「文字表示方式」等がある。

【0014】しかしながら、文書の局所において頻繁に使用される文字と、文書の全般にわたって平均的に使用される文字とがあると、平均的に使用される文字が局所的に使用される文字のためにフォントキャッシュメモリから削除されてしまい、フォントキャッシュメモリの使用効率を十分に上げることができないという問題があった。

【0015】本発明の目的は、前記した従来技術の問題

10

20

30

40

50

点を除去し、フォントキャッシュメモリの再利用率を向上することができるプリント装置を提供することにある。

【0016】

【課題を解決するための手段】上記した目的を達成するために、本発明では、ベクトルフォントをドットパターンに展開して登録するフォントキャッシュメモリを備えた文字処理装置において、文書の予定の範囲を対象に、各文字の出現傾向を予め検出する手段と、前記出現傾向に基づいて、各文字を局部的に出現するローカル文字および全体的に出現するグローバル文字に分類する手段と、ローカル文字のドットパターンを記憶する第1のフォントキャッシュメモリと、グローバル文字のドットパターンを記憶する第2のフォントキャッシュメモリとを具備した点に特徴がある。

【0017】

【作用】上記した構成によれば、各文字が予めローカル文字またはグローバル文字に分類され、それぞれ別々のフォントキャッシュメモリへ登録されるので、ローカル文字が多数登録されることによって、グローバル文字が

【0018】

【実施例】以下に、図面を参照して本発明を詳細に説明する。図1は、本発明の文字処理装置の機能ブロック図である。

【0019】同図において、出現傾向検出手段100は、文書の予定の範囲を対象に、各文字の出現傾向を予め検出する。分類手段200は、前記出現傾向に基づいて、各文字を、文書中の狭い範囲で出現する局部的出現文字（ローカル文字）および広範囲にわたって出現する

全体的出現文字（グローバル文字）に分類する。
【0020】展開手段300は、印字要求のあった文字をベクトルフォント400から読み出して展開し、ローカル文字は第1のフォントキャッシュメモリ（ローカルフォントキャッシュメモリ：LFCM）500Lへ登録し、グローバル文字は第2のフォントキャッシュメモリ（グローバルフォントキャッシュメモリ：GFCM）500Gへ登録する。

【0021】図2は、本発明の一実施例である文字処理装置（プリンタ装置）のハード構成を示すブロック図である。

【0022】図において、ROM2には、当該文字処理装置の全体の動作を制御するプログラム、フォントキャッシュメモリ8を制御するプログラム、電源を入れた直後のプリンタの動作を規定するプログラム、各文字の出現範囲を予め検出するためのプログラム、各文字をローカル文字およびグローバル文字に分類するためのプログラム等が格納されている。RAM3は、CPU1のスタックの退避エリアやワークエリア等に使用される。

【0023】フォントキャッシュメモリ8は、ローカル文字を記憶するローカルフォントキャッシュメモリ部8Lとグローバル文字を記憶するグローバルフォントキャッシュメモリ部8Gとを有する。

【0024】CPU1は、プリンタインタフェース4を通じて、プリンタエンジン5に命令やイメージデータを送信し、逆に、プリンタエンジン5は、該プリンタインタフェース4を通じて、プリンタエンジン内部の状態をCPU1に送信する。プリンタエンジン5は、プリンタインタフェース4を通じて送信されたイメージデータを、実際に紙に印字する。

【0025】前記CPU1は、ホストインタフェース7を介して、ホストワークステーション6から印字すべき文書のデータやプリント制御命令等を受取り、それらをもとに、ベクトルフォントメモリ9に記憶されたベクトル形式のデータをドット形式のデータに展開してページメモリ10に1ページ分のビットマップイメージを形成し、プリンタエンジン5に送出するという動作を制御する。

【0026】次に、前記ホストワークステーション6からCPU1が受取る文書データの構成の一例を、図13を参照して説明する。

【0027】文書データ1ページ分の構成は、位置指定命令と、文書の内容である一字々々に対する命令と、ページ印刷命令等からなっている。位置指令命令は文書の一文字目の印字位置を指示する命令である。前記一字々々に対する命令は、フォント設定命令、文字印字命令等からなっている。また、ページ印刷命令は、文書データの各ページの末尾に設けられている。さらに、最終ページの末尾にはファイル終了記号（図示せず）が設けられている。

【0028】なお、図13には、本発明の動作と関係する命令のみが記されており、他の命令の記載は省略されている。

【0029】次に、図3～図7に示されているフローチャート、および図13の文書データを参照して、文書処理装置のCPU1によるフォントキャッシュメモリ8の制御動作を説明する。

【0030】図3のステップS1において、CPU1はまずホストワークステーション6から文書データを受取る。該文書データは、ページメモリ10の空き領域、又はRAM3に保存される。

【0031】次に、ステップS2へ進んで、文字の出現範囲を調べる単位の設定が行われる。例えば、1文書が該範囲として設定される。この範囲の設定は、どのように行っても良いが、本実施例では、ホストワークステーション6からの指示によって決定するものとする。

【0032】いま、前記ステップS2の範囲設定により、図13に示されている文書データが設定されたとなると、ステップS3では、各文字の出現傾向を求める為

のブレスキャン処理が実行される。具体的には、文書（ファイル）中に出現する全ての文字に関して、文書中の最初の出現場所と最後の出現場所とを求め、これに基づいて各文字をグローバル文字またはローカル文字のいずれかに分類する。

【0033】図4は、当該ブレスキャンを詳細に示したフローチャートである。

【0034】ブレスキャンでは、図4のステップS302において、指定された範囲から1命令が読み込まれる。初めは位置指定命令が入力されるので、ステップS303の判断は否定になり、ステップS306では、フォント設定命令であるか否かの判断がなされる。この判断が否定になると、ステップS308では、文字印字命令か否かの判断がなされる。この実施例では、この判断も否定となり、当該処理はステップS302へ戻って次の1命令が入力される。

【0035】図13から明らかなように、今度はフォント設定命令（12ボ、明朝体）が入力されるので、ステップS303の判断は再び否定になり、ステップS306に進む。今度はステップS306の判断が肯定となるのでステップS307へ進み、カレント・フォントが更新される。すなわち、本例では、カレント・フォント（サイズと書体）が「12ボ、明朝体」に設定される。

【0036】ステップS307の処理が終わるとステップS302に戻り、次の1命令が入力される。本例では、文字印字命令（あ）が入力される。このため、ステップS302に続くステップS303、S306の各判断は否定となり、ステップS308の判断が肯定となってステップS309へ進む。

【0037】ステップS309では、先に読み込まれたカレント・フォント（12ボ、明朝体）および（あ）の文字コードをキーとして、文字出現場所記憶テーブルが検索される。

【0038】ステップS310では、当該文字に関するエリアが文字出現場所記憶テーブルに設定されているか否かの判断がなされる。設定されていない時にはステップS312に進んで、前記カレント・フォント「12ボ、明朝体」と文字コード（あ）に対応するエリアが設定され、“最初の出現場所”を登録するエリアに現在のスキャンアドレス（バイト）が登録される。

【0039】図8は、文字出現場所記憶テーブルの一例を示した図であり、例えばRAM3上に設けられる。文字出現場所記憶テーブルでは、文字ごとに最初の出現場所と最後の出現場所とがバイトで管理されている。図において、「明12あ」は、書体が明朝体でサイズが12ボの文字「あ」を示し、「ゴ8た」は、書体がゴシック体でサイズが8ボの文字「た」を示している。

【0040】なお、実際には、このときの文字出現場所記憶テーブルには「明12あ」のみが登録されるが、図8では、便宜上「明12あ」以外の文字も記録されてい

るように記載している。

【0041】上記の処理が行われると、再度ステップS302に戻り、次の1命令の入力が行われる。ここで、前記ページ印刷命令、フォント設定命令、文字印字命令以外の命令が入力してきた場合には、ステップS303、S306およびS308がいずれも否定となり、ステップS302に戻って次の命令が入力されることとなる。

【0042】ステップS302において、次の文字である文字コード（い）に対するフォント設定が入力されると、前記と同様の動作により、ステップS307でカレント・フォント「12ボ、明朝体」が設定され、ステップS302に戻る。

【0043】ステップS302で文字印字命令（い）が入力されると、ステップS303、S306が否定、S308が肯定になってステップS310に進む。文字出現場所記憶テーブルには、文字コード（い）に対するエリアが確保されていないので、ステップS312に進んで、カレント・フォント「12ボ、明朝体」と文字コード（い）に対応するエリアが設定され、現在のスキャンアドレスが“最初の出現場所”として登録される。

【0044】次に、ステップS302において、3つ目の文字である文字コード（う）に対するフォント設定が入力されると、ステップS307ではカレント・フォント「13ボ、ゴシック体」が設定され、ステップS312では、カレント・フォント「13ボ、ゴシック体」と文字コード（う）に対応するエリアが設定され、そこに現在のスキャンアドレスが最初の出現場所として登録される。

【0045】次いで、ステップS302において、4つ目の文字である文字コード（あ）に対するフォント設定命令が入力されると、ステップS307では再びカレント・フォント「12ボ、明朝体」が設定され、今度はステップS310の判断が肯定となるので、当該処理はステップS311へ進み、文字出現場所記憶テーブルの前記文字（あ）に関する“最後の出現場所”に、現在のスキャンアドレスが登録される。

【0046】なお、フローチャートから明らかなように、“最初の出現場所”は書き換えられることはないが、“最後の出現場所”は再び文字が出現したときに順次書き換えられることになる。

【0047】以上の動作が、1文書分繰返されると、前記文字出現場所記憶テーブルには、該1文書に現れた各文字の“最初の出現場所”と“最後の出現場所”が登録されたことになる。

【0048】文書データの最後の命令であるファイル終了記号が、ステップS302で入力されると、ステップS303の判断は肯定となり、当該処理はステップS304へ進む。ステップS304では、図5のフローチャートに詳細を示したように、各文字をグローバル文字ま

たはローカル文字のいずれかに分類するための分類処理が実行される。

【0049】分類処理がスタートすると、ステップS3042では、前記文字出現場所記憶テーブルを用いて各文字の出現範囲を求める。この出現範囲は、“最後の出現場所”から“最初の出現場所”を差し引くことによって求められる。

【0050】ステップS3043では、RAM3上に設けられたグローバル文字テーブルに空き領域があるかが判定され、空き領域があると、ステップS3044では、前記出現範囲の最も広い文字がグローバル文字テーブル(図9参照)へ登録されると共に、当該文字が文字出現場所記憶テーブルから削除される。このグローバル文字テーブルも、RAM3上に設けることができる。

【0051】以後、前記と同様にして、残りの文字を対象に、出現範囲が最も広い文字のグローバル文字テーブルへの登録が行われる。グローバル文字テーブルの空き領域がなくなると、当該処理は図3のステップS4へ進み、CPU1はホストワークステーション6から文書データを受取ってビットマップイメージに展開する処理を

実行する。

【0052】展開処理の一例を、図6のフローチャートを参照して説明する。

【0053】展開処理がスタートすると、ステップS402では、1命令の読み込みが行われる。ステップS403、S405およびS407では、読込まれた命令がページ印刷記号か、フォント設定命令か、文字印字命令かの判断がなされる。

【0054】ページ印刷記号の場合には、ステップS403が肯定となってメインルーチンのステップS5へ進む。フォント設定命令の場合には、ステップS405が肯定となってステップS406に進み、カレントフォントを更新する動作が行われる。さらに、文字印字命令であった場合には、ステップS407が肯定となってステップS408へ進み、文字コードおよびカレントフォントをキーにしてグローバル文字テーブルを検索する動作が行われる。

【0055】一方、前記の三つの命令以外の命令であった場合には、ステップS409へ進んで、その命令に回答した処理が実行される。

【0056】CPU1が前記文字印字命令を読込んだ場合には、前記ステップS408においてグローバル文字テーブルが検索され、ステップS410では、当該文字がグローバル文字テーブルに登録されているか否かの判断が行われる。

【0057】登録されていると判定されると、ステップS411では、文字コードおよびカレントフォントをキーにしてグローバルフォントキャッシュメモリ部8G(以下、GFCMと表現する場合もある)が検索される。ステップS412において、GFCMに登録されて

いないと判断されると、ステップS413では、GFCMに登録可能か否かが判断され、登録不可能であると、ステップS414では、LRU方式によりGFCMの文字が削除される。

【0058】また、ステップS413において、登録可能であると判断されると、ステップS415では、ベクトルフォントが展開されてGFCMへ登録される。ステップS416では、展開された文字がページメモリへ出力される。

10 【0059】一方、前記ステップS410において、グローバル文字ではないと判定されると、ステップS417では、文字コードおよびカレントフォントをキーにしてローカルフォントキャッシュメモリ部8L(以下、LFCMと表現する場合もある)が検索される。ステップS418において、LFCMに登録されていないと判断されると、ステップS419では、LFCMに登録可能か否かが判断され、登録不可能であると、ステップS420では、LRU方式によりLFCMの文字が削除される。

20 【0060】また、ステップS419において、登録可能であると判断されると、ステップS421では、ベクトルフォントが展開されてLFCMへ登録される。ステップS422では、展開された文字がページメモリへ出力される。

【0061】本実施例によれば、各文字を、文書中の狭い範囲で出現するローカル文字および広範囲にわたって出現するグローバル文字に分類し、それぞれを別のキャッシュメモリへ登録するようにしたので、文書中にローカル文字とグローバル文字が混在しても、グローバル文字がローカル文字によってフォントキャッシュメモリ上から追い出されてしまうことがなく、フォントキャッシュメモリの再利用率を向上することができる。

30 【0062】次に、本発明の第2実施例を、図11、図12のフローチャートを参照して説明する。なお、前記と同一の符号を付したステップでは同一または同等の処理が実行されるので、その説明は省略する。

【0063】上記した第1実施例では、各文字をグローバル文字またはローカル文字に分類する基準を、“最初の出現場所”と“最後の出現場所”との差である出現範囲としていたが、本実施例では、各文字が出現したページ数に基づいて各文字を分類するようにした点に特徴がある。

40 【0064】プレスキャンが開始されると、図11に示したように、ステップS301ではページカウンタが初期化される。その後、ステップS302において文字印字命令が読み込まれ、ステップS310において、当該文字に関するエリアが出現ページ数記憶テーブルに設定されているか否かの判断がなされる。

【0065】図10は、出現ページ数記憶テーブルの一例を示した図であり、例えばRAM3上に設けられる。

出現ページ数記憶テーブルでは、文字ごとに“最終出現ページ”と“出現ページ数”が登録される。

【0066】設定されていない時にはステップS312に進んで、前記カレント・フォント「12ボ、明朝体」と文字コード（あ）に対応するエリアが設定され、“最終出現ページ”を登録するエリアに前記ページカウンタの値がセットされ、“出現ページ数”を登録するエリアに“1”がセットされる。

【0067】一方、ステップS310において、当該文字に関するエリアが出現ページ数記憶テーブルに設定されていると判定されると、ステップS316では、その最終出現ページがページカウンタの値と比較され、両者が一致していないと、ステップS311において、最終出現ページがページカウンタの値に更新されると共に、“出現ページ数”がインCREMENTされる。

【0068】その後、文書データの最後の命令であるファイル終了記号が、ステップS302で入力されると、ステップS303の判断は肯定となり、当該処理はステップS304aへ進む。ステップS304aでは、図12のフローチャートに詳細を示したように、各文字をグ

ローバル文字またはローカル文字のいずれかに分類するための分類処理が実行される。

【0069】分類処理がスタートすると、ステップS3043では、RAM3上に設けられたグローバル文字テーブルに空き領域があるか否かが判定され、空き領域があると、ステップS3044aでは、“出現ページ数”の最も大きい文字がグローバル文字テーブルへ登録されると共に、当該文字が出現ページ数記憶テーブルから削除される。

【0070】本実施例によっても、各文字を、文書中の狭い範囲で出現するローカル文字および広範囲にわたって出現するグローバル文字に分類し、それぞれを別のキャッシュメモリへ登録するようにしたので、フォントキャッシュメモリの再利用率を向上することができる。

【0071】なお、上記した各実施例では、文書の展開前にプリンタ側でプレスキャン行って各文字の出現傾向を調べ、各文字をローカル文字またはグローバル文字に分類するものとして説明したが、本発明はこれのみに限定されず、例えば平仮名の“は”、“や”などのような頻出文字を予めグローバル文字として登録しておくようにすれば、プリンタ側で文書データの解析を行う必要がなくなり、プリンタの負担を軽減することができる。

【0072】

【発明の効果】以上の説明から明らかなように、本発明によれば、各文字を、文書中の狭い範囲で出現するローカル文字および広範囲にわたって出現するグローバル文字に分類し、それぞれを別のキャッシュメモリへ登録するようにしたので、文書中にローカル文字とグローバル文字が混在しても、グローバル文字がローカル文字によってフォントキャッシュメモリ上から追い出されてしまうことがなく、フォントキャッシュメモリの再利用率を向上することができる。

10 【図面の簡単な説明】

【図1】 本発明の機能ブロック図である。

【図2】 本発明の一実施例である文書処理装置のブロック図である。

【図3】 本発明の第1実施例の動作を説明するためのフローチャートである。

【図4】 図3のフローチャートのステップS3の詳細を示すフローチャートである。

【図5】 図4のフローチャートのステップS304の詳細を示すフローチャートである。

20 【図6】 図3のフローチャートのステップS4の詳細を示すフローチャート（その1）である。

【図7】 図3のフローチャートのステップS4の詳細を示すフローチャート（その2）である。

【図8】 文字出現場所記憶テーブルの構造を模式的に示した図である。

【図9】 グローバル文字テーブルの構造を模式的に示した図である。

【図10】 出現ページ数記憶テーブルの構造を模式的に示した図である。

30 【図11】 第2実施例のプレスキャン処理を説明するためのフローチャートである。

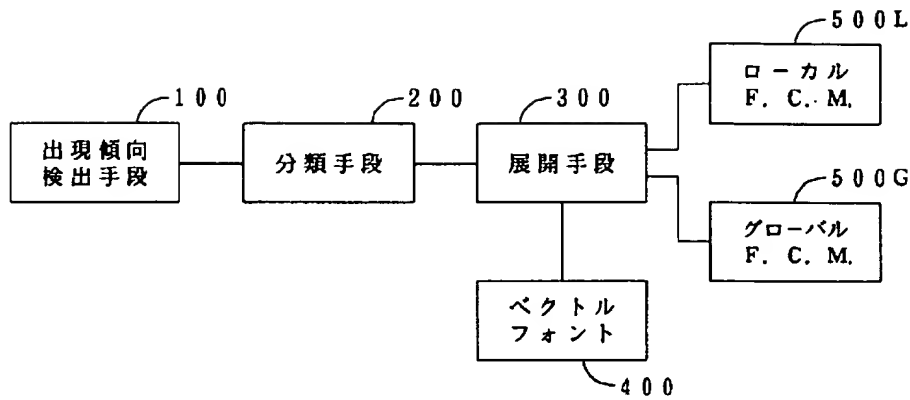
【図12】 第2実施例の文字分類処理を説明するためのフローチャートである。

【図13】 文書データの概要を示す図である。

【符号の説明】

1…CPU、2…ROM、3…RAM、4…プリンタI/F、5…プリンタエンジン、6…ホストW/S、7…ホストI/F、8、500…フォントキャッシュメモリ、9…ベクトルフォントメモリ、10…ページメモリ、100…出現傾向検出手段、200…分類手段、300…展開手段、400…ベクトルフォント、500G…グローバルフォントキャッシュメモリ、500L…ローカルフォントキャッシュメモリ

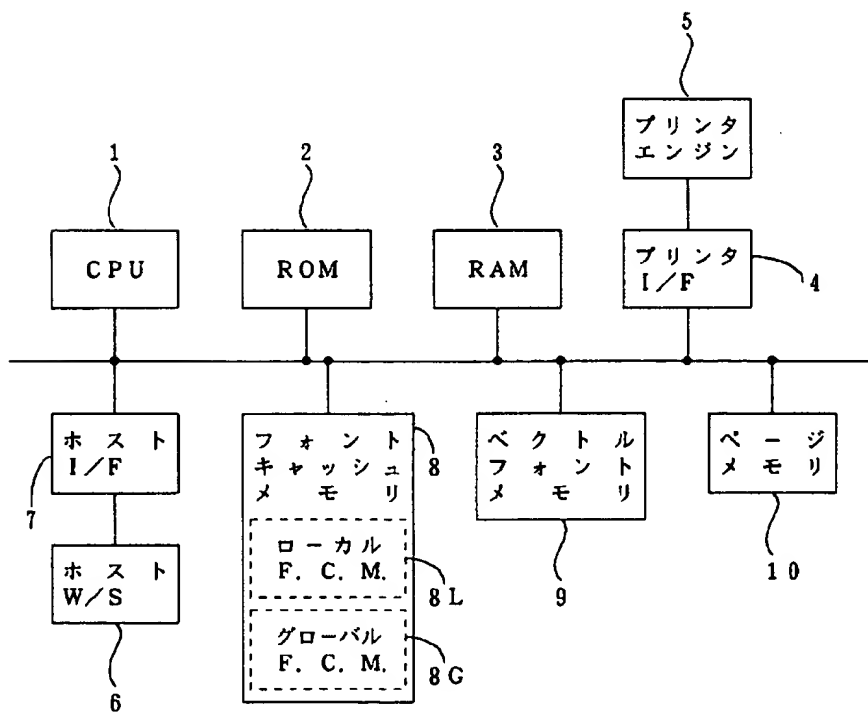
【図1】



【図9】

グローバル文字
ゴ16あ
明12い
明12と

【図2】



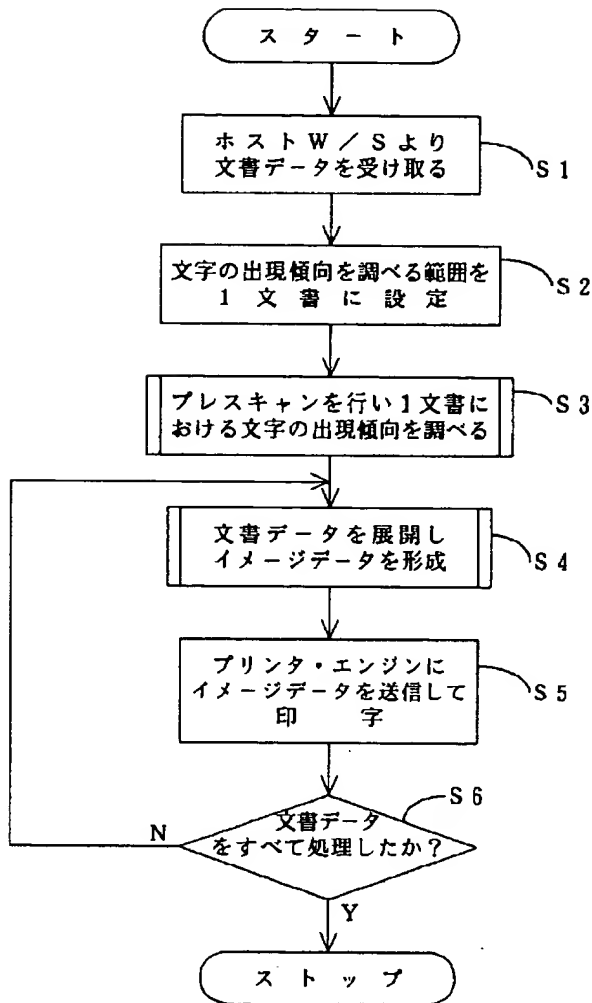
【図8】

文 字	最初の出現場所 (バイト)	最後の出現場所 (バイト)
明12あ	256	1024
明12い	512	10256
ゴ8た	12500	13512
ゴ8か	14500	16024

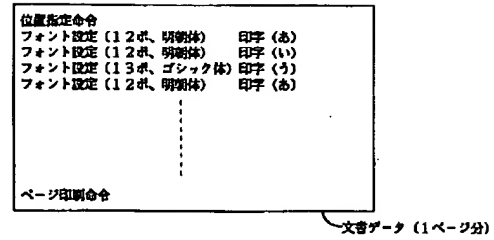
【図10】

文 字	最終出現ページ	出現ページ数
明12A	7	5
ゴ18あ	12	11
ゴ8た	10	1
ゴ8か	5	10

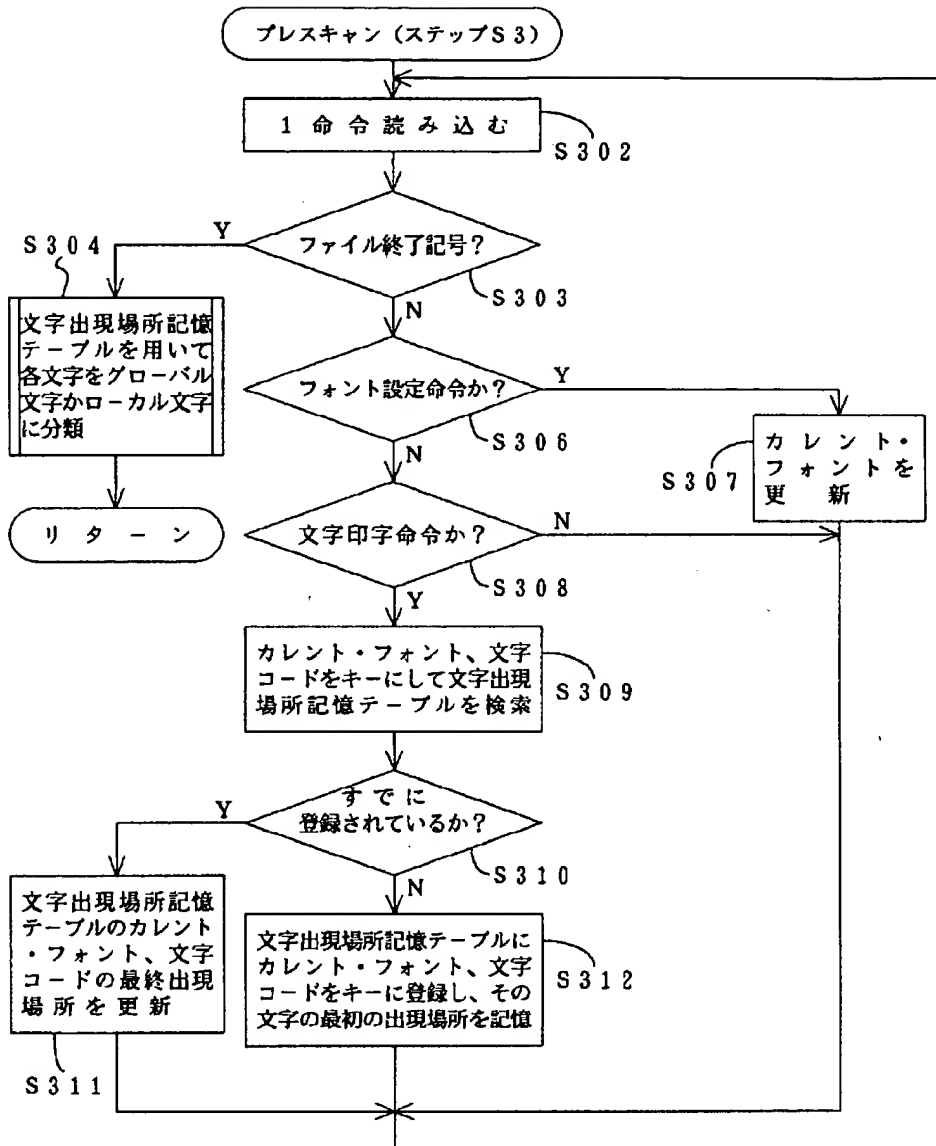
【図3】



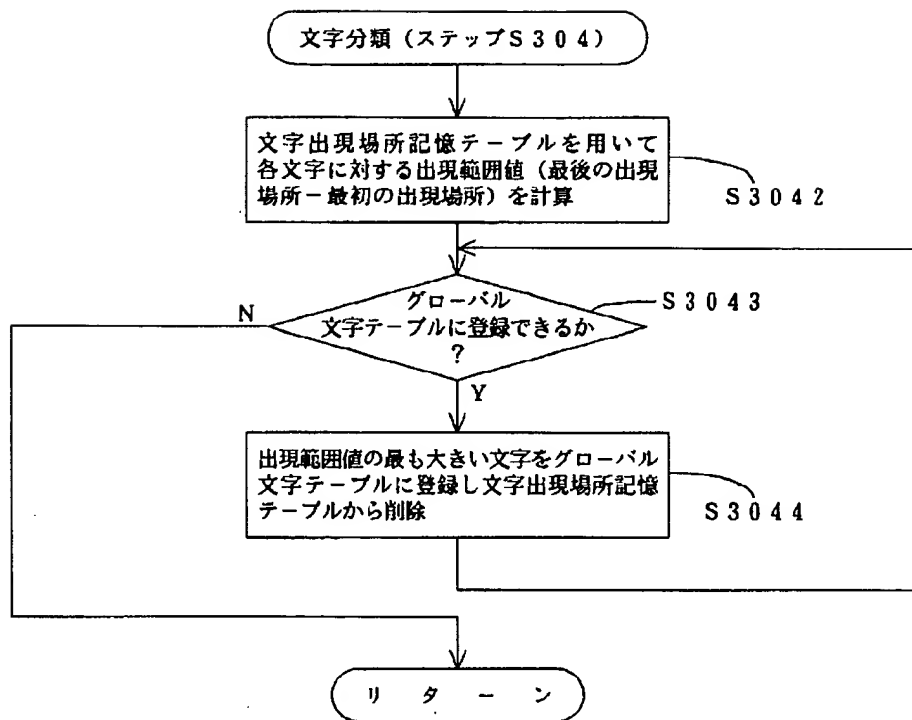
【図13】



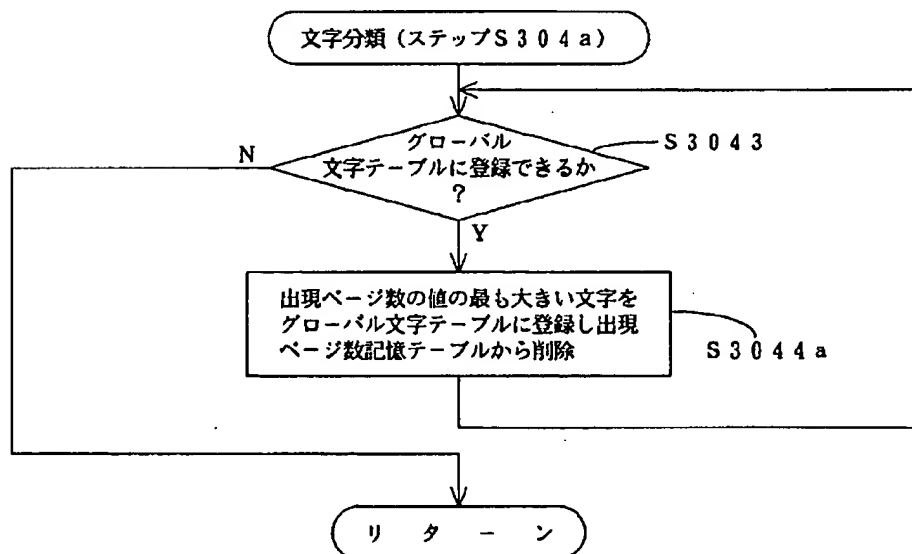
【図4】



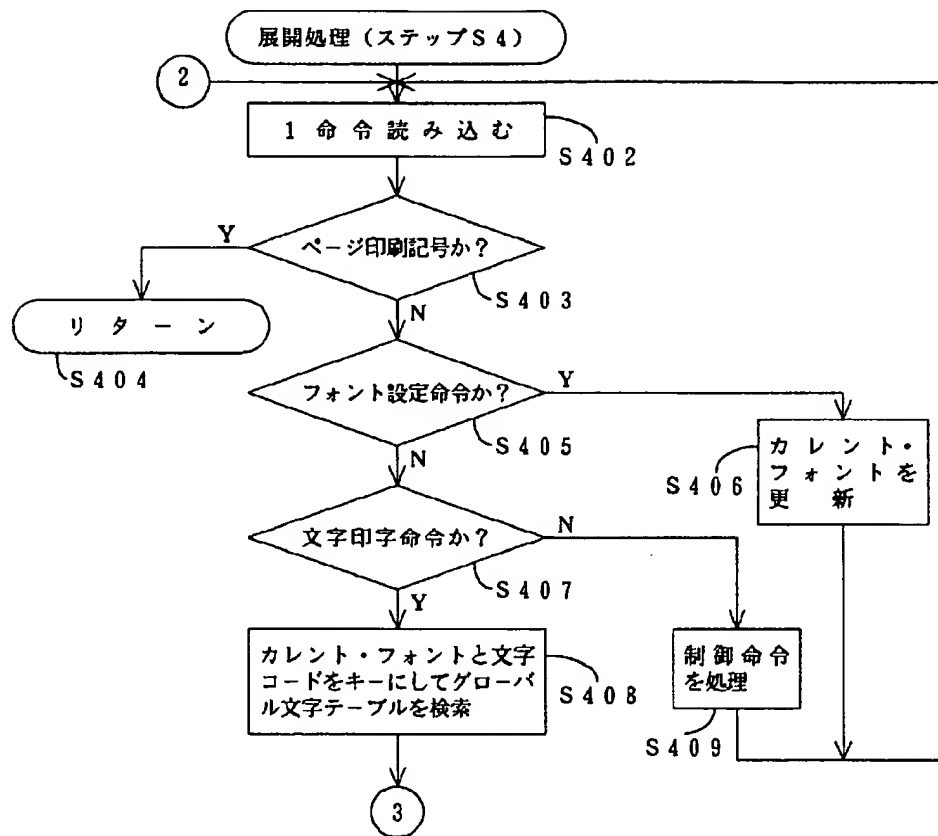
【図5】



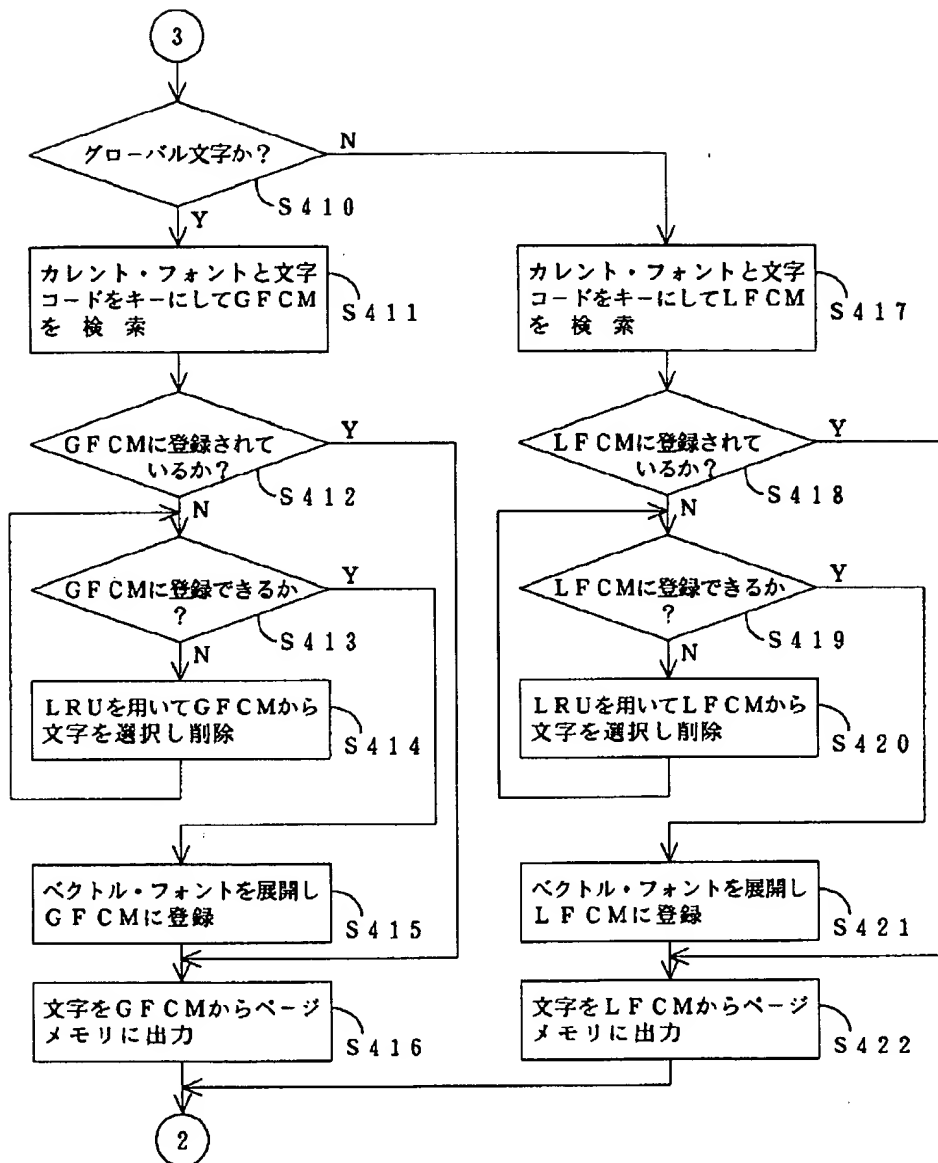
【図12】



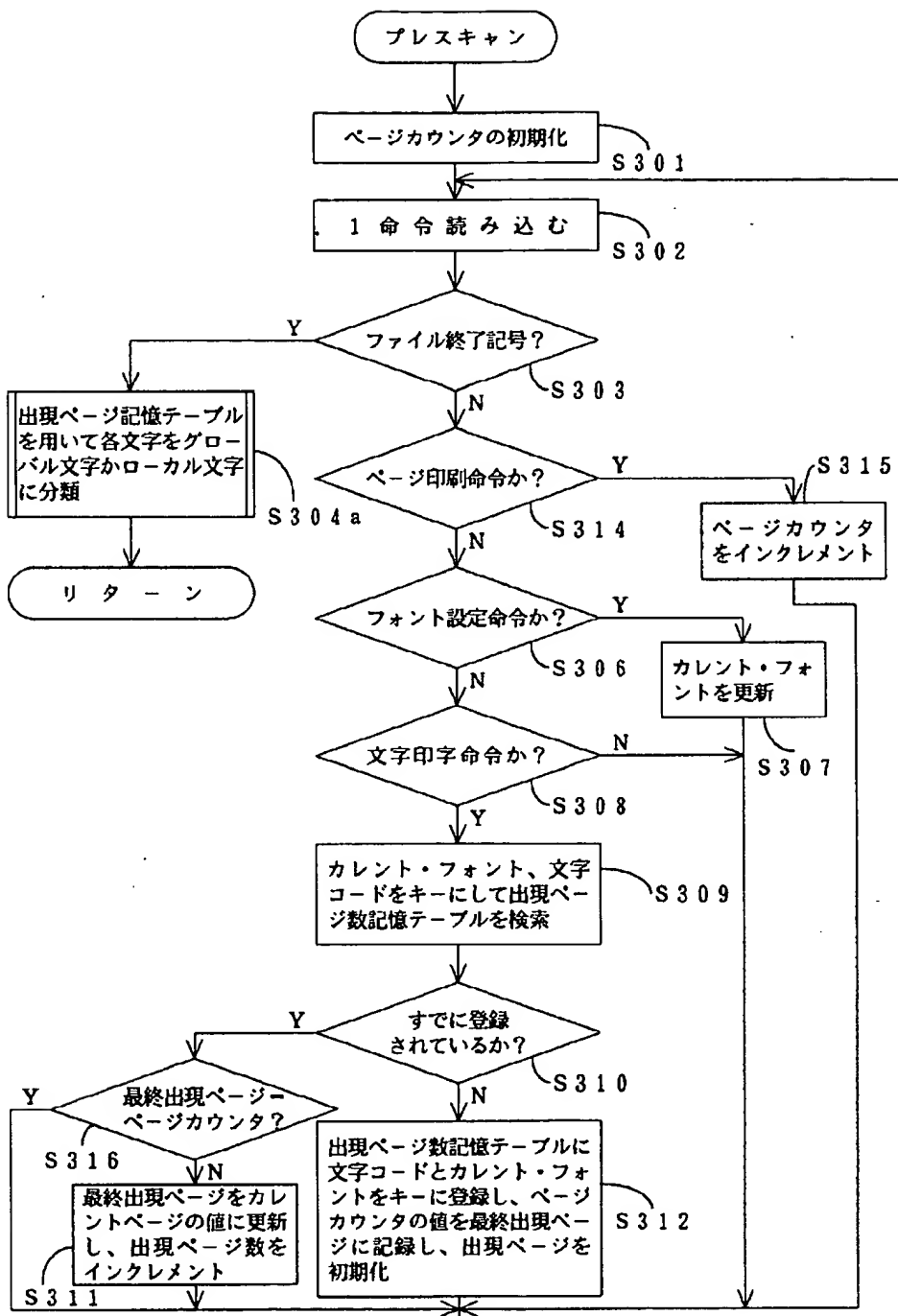
【図6】



【図7】



【図11】



PATENT ABSTRACTS OF JAPAN

(11)Publication number : 06-183062

(43)Date of publication of application : 05.07.1994

(51)Int.Cl.

B41J 2/485

B41J 5/44

G09G 5/22

(21)Application number : 04-353891

(71)Applicant : FUJI XEROX CO LTD

(22)Date of filing : 16.12.1992

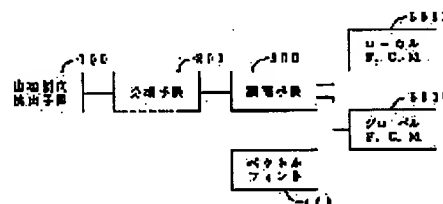
(72)Inventor : KASHIMA YOZO

(54) CHARACTER PROCESSING DEVICE

(57)Abstract:

PURPOSE: To largely improve use efficiency of font cache memory.

CONSTITUTION: An appearance tendency detecting means 100 detects previously appearance tendency of characters in a scheduled range of a document. A classifying means 200 classifies the characters based on the appearance tendency into local characters which appear in a narrow range of the document and global characters which appear in a wide range. A developing means 300 reads characters, which are required to be printed, from a vector font 400 and develops them, and the local characters are registered in a local font cache memory 500L and the global characters are registered in a global font cache memory 500G.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office